This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Please amend the claims as follows:

1. (Currently Amended) A method of transmitting information between a first computer and a second computer over a network, comprising the steps of:

(1) embedding in a header of each of a plurality of data packets a network address ~~discriminator value~~ that periodically changes between successive data packets, wherein each network address is used to route packets over the network ~~discriminator value is not based solely on the value of other data in each data packet~~;

(2) transmitting the plurality of data packets between the first computer and the second computer;

(3) receiving the transmitted data packets at the second computer; and

(4) for each received data packet, comparing the network address ~~discriminator value~~ to a moving window of valid network addresses ~~set of valid discriminator values~~ and, in response to detecting a match within the moving window, accepting the received data packet for further processing, and otherwise rejecting the received data packet.

2. (Currently Amended) The method of claim 1, wherein step (1) comprises the step of using an Internet Protocol address in an Internet Protocol header as the network address ~~discriminator value~~, wherein the Internet Protocol address is used to route the data packets over the Internet.

3. (Canceled)

4. (Currently Amended) The method of claim 1, further comprising the step of embedding ~~using~~ an additional quasi-random value in ~~as the discriminator value~~ a data field external to an Internet Protocol header of each data packet.

5. (Original) The method of claim 1, wherein steps (1) and (4) are performed in a data link layer of an ISO standard communication protocol.

6. (Currently Amended) The method of claim 1, wherein step (1) comprises the step of using a Media Access Control (MAC) hardware address as the network address ~~discriminator value~~, wherein the MAC hardware address is used to route the data packets on a local area network.

7. (Currently Amended) The method of claim 1, wherein step (1) comprises the step of using a different network address ~~discriminator value~~ for each successive packet.

8. (Currently Amended) The method of claim 1, ~~wherein step (4) comprises the step of comparing each discriminator value to a window of valid discriminator values, wherein the window is wide enough to permit comparison to only a small number of potentially valid discriminator values, and~~ further comprising the step of moving the window as each successive data packet is ~~packets are~~ received.

9. (Currently Amended) The method of claim 1, further comprising the step of sharing between the first computer and the second computer information sufficient to generate the set of valid network addresses ~~discriminator values~~.

10. (Currently Amended) The method of claim 1, further comprising the step of transmitting from the first computer to the second computer an algorithm for selecting successively valid network addresses ~~discriminator values~~.

11. (Original) The method of claim 1, wherein step (4) comprises the step of using a presence vector to determine whether to accept each data packet.

12. (Currently Amended) The method of claim 1, wherein step (4) comprises the step of using a hashing function to determine whether the network address ~~discriminator value~~ is valid.

13. (Currently Amended) The method of claim 1, further comprising the step of transmitting a synchronization request between the first computer and the second computer, wherein the second computer uses the synchronization request to maintain synchronization of valid network addresses ~~discriminator values~~.

14. (Original) The method of claim 13, further comprising the step of, in response to failure to receive a synchronization acknowledgement from the second computer, shutting off transmission of data packets to the second computer.

15. (Currently Amended) The method of claim 13, further comprising the step of embedding a synchronization value in each data packet that permits the second computer to re-establish synchronization in a set of potentially valid network addresses ~~discriminator values~~.

16. (Currently Amended) The method of claim 13, further comprising the step of moving the ~~a~~ window of valid network addresses ~~discriminator values~~ in the second computer in response to receiving the synchronization request from the first computer.

17. (Currently Amended) The method of claim 1, wherein step (1) comprises the steps of embedding a periodically-changing ~~using an~~ Internet Protocol source address in an Internet Protocol header ~~as a first part of the discriminator value~~ and embedding a periodically-changing ~~using an~~ Internet Protocol destination address in the Internet Protocol header ~~as a second part of the discriminator value~~, wherein the source and destination addresses are used to route each data packet over the Internet.

18. (Original) The method of claim 17, further comprising the steps of:

embedding a plurality of the data packets into a frame; and

embedding a source and destination hardware address in the frame, wherein the source and destination hardware address are quasi-randomly generated and used to route the frame on a network.

19. (Currently Amended) The method of claim 1, further comprising the step of maintaining in the first computer a first transmit table and a first receive table, and maintaining in the second computer a second transmit table and a second receive table,

wherein each transmit table comprises a list of valid network addresses ~~discriminator values~~ that are to be inserted into outgoing data packets;

wherein each receive table comprises a list of valid network addresses ~~discriminator values~~ that are to be compared against incoming data packets; and

wherein the first transmit table in the first computer matches the second receive table in the second computer; and wherein the first receive table in the first computer matches the second transmit table in the second computer.

20. (Currently Amended) A method of transmitting data packets over a network comprising a plurality of computers connected to each other through a plurality of physical transmission paths, the method comprising the steps of:

(1) for each of a plurality of data packets, randomly selecting one of the plurality of physical transmissions paths through the plurality of computers;

(2) selecting a next pair of source and destination network addresses generated from an algorithm that generates a plurality of pairs of source and destination network addresses each associated with the one randomly selected physical transmission path; and

~~(2)~~ (3) transmitting each data packet over the randomly selected physical transmission path using the selected next pair of source and destination network addresses.

21. (Canceled)

22. (Original) The method of claim 20 ~~21~~ wherein step (1) ~~(2)~~ comprises the step of avoiding selection of a path that is not operational.

23. (Currently Amended) A system comprising:

a first computer that embeds into each of a plurality of data packets a network address ~~discriminator value~~ that periodically changes between successive data packets, wherein each network address is used to route packets over a network ~~discriminator value is not based solely on the value of other data in each data packet~~; and

a second computer coupled to the first computer through the ~~a~~ network,

wherein the first computer transmits the plurality of data packets to the second computer, and

wherein the second computer receives the transmitted data packets, compares the network address ~~discriminator value~~ in each received data packet to a ~~set of valid~~ moving window of valid network addresses ~~discriminator values~~ and, in response to detecting a match, accepts the received data packet for further processing, and otherwise rejects the received data packet.

24. (Currently Amended) The system of claim 23, wherein the first computer embeds into each of the plurality of data packets an Internet Protocol address in an Internet Protocol header as the network address ~~discriminator value~~, wherein the Internet Protocol address is used to route the data packets over the Internet.

25. (Canceled)

26. (Currently Amended) The system of claim 23, wherein the first computer embeds an additional quasi-random value ~~the discriminator value~~ in a data field external to an Internet Protocol header of each data packet.

27. (Currently Amended) The system of claim 23, wherein the first computer embeds each network address ~~discriminator value~~ in a first data link layer of an ISO standard communication protocol, and wherein the second computer compares each network address ~~discriminator value~~ in a second data link layer of the ISO standard communications protocol.

28. (Currently Amended) The system of claim 23, wherein the first computer embeds a Media Access Control (MAC) hardware address as the network address ~~discriminator value~~, wherein the MAC hardware address is used to route the data packets on a local area network.

29. (Currently Amended) The system of claim 23, wherein the first computer embeds a different network address ~~discriminator value~~ for each successive packet.

30. (Currently Amended) The system of claim 23, wherein the second computer ~~compares each discriminator value to a window of valid discriminator values, wherein the window is wide enough to permit comparison to only a small number of potentially valid discriminator values, and wherein the window is moved~~ moves the window as each successive data packet is ~~packets are~~ received.

31. (Currently Amended) The system of claim 23, wherein the first and second computers share common information sufficient to generate the set of valid network addresses. ~~discriminator values.~~

32. (Currently Amended) The system of claim 23, wherein the first computer transmits to the second computer an algorithm for selecting successively valid network addresses. ~~discriminator values.~~

33. (Original) The system of claim 23, wherein the second computer uses a presence vector to determine whether to accept each data packet.

34. (Currently Amended) The system of claim 23, wherein the second computer uses a hashing function to determine whether the network address ~~discriminator value~~ is valid.

35. (Currently Amended) The system of claim 23, wherein the first computer transmits to the second computer a synchronization request, wherein the second computer uses the synchronization request to maintain synchronization of valid network addresses ~~discriminator values~~.

36. (Original) The system of claim 35, wherein the first computer, in response to failure to receive a synchronization acknowledgement from the second computer, shuts off transmission of data packets to the second computer.

37. (Currently Amended) The system of claim 35, wherein the first computer embeds a synchronization value in each data packet that permits the second computer to re-establish synchronization in a set of potentially valid network addresses ~~discriminator values~~.

38. (Currently Amended) The system of claim 35, wherein the second computer moves a window of valid network addresses ~~discriminator values~~ in response to receiving the synchronization request from the first computer.

39. (Currently Amended) The system of claim 23, wherein the first computer embeds a periodically-changing ~~an~~ Internet Protocol source address in an Internet Protocol header ~~as a first part of the discriminator value~~ and embeds a periodically-changing ~~an~~ Internet Protocol destination address in the Internet Protocol header ~~as a second part of the discriminator value~~, wherein the source and destination addresses are used to route each data packet over the Internet.

40. (Original) The system of claim 39, wherein the first computer embeds a plurality of the data packets into a frame and embeds a source and destination hardware address in the frame, wherein the source and destination hardware address are quasi-randomly generated and used to route the frame on a network.

41. (Currently Amended) The system of claim 23,

wherein the first computer comprises a first transmit table and a first receive table,

wherein the second computer comprises a second transmit table and a second receive table,

wherein each transmit table comprises a list of valid network addresses ~~discriminator values~~ that are to be inserted into outgoing data packets,

wherein each receive table comprises a list of valid network addresses ~~discriminator values~~ that are to be compared against incoming data packets,

wherein the first transmit table in the first computer matches the second receive table in the second computer, and

wherein the first receive table in the first computer matches the second transmit table in the second computer.

42. (Currently Amended) A router ~~first computer~~ coupled to a network comprising a plurality of computers connected to each other through a plurality of physical transmission paths,

wherein the router ~~first computer~~ receives ~~generates~~ a plurality of data packets for transmission across the network; and

wherein the router ~~first computer~~, for each data packet ~~of a plurality of data packets~~, randomly selects one of the plurality of physical transmission paths through the plurality of computers and transmits each data packet over the randomly selected physical transmission

path using a pair of source and destination network addresses generated from an algorithm that generates a plurality of pairs of source and destination addresses each associated with the one randomly selected physical transmission path.

43. (Canceled)

44. (Currently Amended) The router ~~first computer~~ of claim 42 ~~43~~, wherein the router avoids selection of a non-operational path ~~selected path is operational~~.

45. (Currently Amended) A system comprising in combination:

a transmitting node that generates pseudo-random network addresses ~~discriminator values~~ and embeds the pseudo-random network addresses ~~discriminator values~~ into headers of data packets for transmission; and

a receiving node that receives data packets transmitted by the transmitting node, wherein the receiving node, for each received packet, extracts each ~~the~~ pseudo-randomly generated network address ~~discriminator value~~, compares it to a moving window ~~set~~ of potentially valid network addresses ~~discriminator values~~ shared between the transmitting node and the receiving node and, in response to detecting a match, accepts the data packet, and otherwise discards the packet.

46. (Currently Amended) The system of claim 45, wherein the receiving node maintains a window of valid network addresses ~~discriminator values~~, wherein the window is moved in response to detecting a match.

47. (Currently Amended) The system of claim 45, wherein each pseudo-randomly generated network address ~~discriminator value~~ comprises a valid Internet Protocol address that is assigned to the receiving node.

48. (Currently Amended) The system of claim 45, wherein each pseudo-randomly generated network address ~~discriminator value~~ comprises a valid Media Access Control (MAC) hardware address that is assigned to the receiving node.

49. (Currently Amended) The system of claim 45, wherein the transmitting node generates a different pseudo-randomly generated network address ~~discriminator value~~ for each successive data packet.

50. (Currently Amended) A receiving computer that receives data packets from a transmitting computer, wherein the receiving computer comprises computer instructions that execute the steps of:

(1) for each received data packet, extracting a discriminator value inserted by the transmitting computer;

(2) comparing the extracted discriminator value to a set of valid discriminator values on the basis of information previously shared with the transmitting computer; and

(3) in response to detecting a match in step (2), accepting the received data packet for further processing and otherwise rejecting the data packet, wherein the receiving computer maintains a sliding window of valid discriminator values, wherein the window slides to encompass a next range of valid discriminator values in response to detecting matches.

51. (Original) The receiving computer of claim 50, wherein the receiving computer further comprises computer instructions that extract as the discriminator value an Internet Protocol address from a header portion of each data packet.

52. (Canceled).

53. (Original) The receiving computer of claim 50, wherein the receiving computer receives information from the transmitting computer sufficient to establish the set of valid discriminator values.

54. - 67. (Canceled)

68. (Currently Amended) A transmitting computer that transmits data packets to a receiving computer over a network, wherein the transmitting computer comprises computer instructions that execute the step of, for each transmitted data packet, inserting into a header of the data packet a network address ~~discriminator value~~ for extraction by the receiving computer, wherein the network address ~~discriminator value~~ is used to route data packets over the network and is generated using an algorithm that selects the network address quasi-randomly from a plurality of network addresses that are each mapped to the receiving computer. ~~on the basis of information previously shared with the receiving computer;~~

69. (Currently Amended) The transmitting computer of claim 68, wherein the transmitting computer further comprises computer instructions that insert as the network address ~~discriminator value~~ an Internet Protocol address into the ~~a~~ header portion of each data packet.

70. (Currently Amended) The transmitting computer of claim 68, wherein the transmitting computer transmits information to the receiving computer sufficient to establish a ~~the~~ set of valid network addresses ~~discriminator values~~.

71. - 84. (Canceled)

85. (New) A method of communicating between a first node and a second node coupled via a network, comprising the steps of:

(1) in the first node, using a first algorithm to select from among a first plurality of different network addresses each of which is mapped in the network to the first node, and using

each selected network address in a header of a packet that is transmitted over the network to the second node; and

(2) in the second node, using a second algorithm to select from among a second plurality of network addresses each of which is mapped in the network to the second node, and using each selected network address in a header of a packet that is transmitted over the network to the first node.

86. (New) The method of claim 85, wherein the network comprises an Internet Protocol (IP) network, and wherein each network address comprises an Internet Protocol (IP) source or destination address.

87. (New) The method of claim 86, wherein:

step (1) comprises the step of using each selected IP address as the destination address of the second node; and

step (2) comprises the step of using each selected IP address as the destination address of the first node.

88. (New) The method of claim 86, wherein:

step (1) comprises the step of using each selected IP address as the source address of the first node; and

step (2) comprises the step of using each selected IP address as the source address of the second node.

89. (New) The method of claim 85, further comprising the steps of:

(3) in the first node, evaluating a header of received packets to determine whether it contains a network address that conforms to the second algorithm;

(4) in the first node, upon determining that the header contains a network address that conforms to the second algorithm, accepting the packet for processing; and

(5) in the first node, upon determining that the header does not contain a network address that conforms to the second algorithm, rejecting the packet for processing.

90. (New) The method of claim 85, wherein the first and second algorithms select each network address on a quasi-random basis.

91. (New) The method of claim 85, wherein the first and second algorithms select a different network address for each outgoing packet.

92. (New) A method of communicating between first and second nodes in a network, comprising the steps of:

(1) in the first node, storing a transmit netblock comprising a plurality of pairs of source and destination IP addresses that will be used for communicating with the second node, and an algorithm for selecting pairs of source and destination IP addresses from among the plurality of pairs of source and destination IP addresses;

(2) in the first node, generating a plurality of IP packets each comprising one of the selected pairs or source and destination IP addresses; and

(3) in the first node, transmitting each IP packet generated in step (2) to the second node.

93. (New) The method of claim 92, wherein the algorithm selects a different pair of source and destination addresses for each IP packet transmitting in step (3).

94. (New) The method of claim 92, further comprising the step of receiving the transmit netblock and the algorithm from the second node.

95. (New) The method of claim 92, further comprising the step of receiving a plurality of IP packets from the second node and, for each received IP packet:

determining whether the received IP packet contains a valid source and destination IP address, wherein the validity of each address is determined with reference to a second algorithm;

upon determining that the received IP packet contains a valid source and destination IP address, accepting the IP packet for further processing; and

upon determining that the received IP packet does not contain a valid source and destination IP address, rejecting the IP packet.

96. (New) The method of claim 95, wherein the second algorithm determines whether each address is valid by generating a range of predictions encompassing a plurality of possible transmitted source and destination addresses, and comparing each address to the range of predictions.

97. (New) The method of claim 96, further comprising the step of discarding received IP packets containing an IP address that was previously received.